

Krupp

A Verification Plan for Systematic Verification of Mechatronic Systems



C-LAB Publication

Herausgegeben von
Published by

Dr. Wolfgang Kern, Siemens AG
Prof. Dr. Franz-Josef Rammig, Universität Paderborn

Das C-LAB - Cooperative Computing & Communication Laboratory - leistet Forschungs- und Entwicklungsarbeiten und gewährleistet deren Transfer an den Markt. Es wurde 1985 von den Partnern Nixdorf Computer AG (nun Siemens AG) und der Universität Paderborn im Einvernehmen mit dem Land Nordrhein-Westfalen gegründet.

Die Vision, die dem C-LAB zugrunde liegt, geht davon aus, daß die gewaltigen Herausforderungen beim Übergang in die kommende Informations- und Wissensgesellschaft nur durch globale Kooperation und in tiefer Verzahnung von Theorie und Praxis gelöst werden können. Im C-LAB arbeiten deshalb Mitarbeiter von Hochschule und Industrie unter einem Dach in einer gemeinsamen Organisation an gemeinsamen Projekten mit internationalen Partnern eng zusammen.

C-LAB - the Cooperative Computing & Cooperation Laboratory - works in the area of research and development and safeguards its transfer into the market. It was founded in 1985 by Nixdorf Computer AG (now Siemens AG) and the University of Paderborn under the auspices of the State of North-Rhine Westphalia.

C-LAB's vision is based on the fundamental premise that the gargantuan challenges thrown up by the transition to a future information and knowledge society can only be met through global cooperation and deep interworking of theory and practice. This is why, under one roof, staff from the university and from industry cooperate closely on joint projects within a common research and development organization together with international partners. In doing so, C-LAB concentrates on those innovative subject areas in which cooperation is expected to bear particular fruit for the partners and their general well-being.

C-LAB Publication

Band 29

Alfred Alexander Krupp

**A Verification Plan for Systematic Verification
of Mechatronic Systems**

D 466 (Diss. Universität Paderborn)

Shaker Verlag
Aachen 2009

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <http://dnb.d-nb.de>.

Zugl.: Paderborn, Univ., Diss., 2009

Copyright Shaker Verlag 2009

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN 978-3-8322-8251-6

ISSN 1438-3527

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen

Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9

Internet: www.shaker.de • e-mail: info@shaker.de

A Verification Plan for Systematic Verification of Mechatronic Systems

Dissertation

A thesis submitted to the Faculty of Computer Science, Mathematics and
Electrical Engineering of Paderborn University in partial fulfilment of the
requirements for the degree of Dr. rer. nat.

by
Alfred Alexander Krupp

Paderborn, 2009

Supervisors:

Prof. Dr. Franz Josef Rammig, University of Paderborn

Prof. Dr. Wolfram Hardt, Chemnitz University of Technology

Date of public examination: 13.03.2009

CVSSTAMP: 1245683775

Abstract

Today, verification of mechatronic systems has become a major cost factor in mechatronic system development. Yet the prevalence of model-based development opens new opportunities for automatized verification. While code-generation from models removes many sources of coding errors, it cannot detect design flaws in the model. Extensive functional verification of mechatronic models has become a necessity. Current mechatronic system verification approaches exhibit a major gap between requirement definition and formal property definition. Besides lack of support for natural language formalization, a standardized and accepted means for formal property definition does not exist as a target for verification planning.

In comparison, in the domain of electronic design the concept of model-based engineering across several levels of abstraction has been employed for several decades. Formal verification, simulation and testing are employed on a regular basis. The increasing demand for verification at a high level of abstraction has led to the definition of methods and languages for a functional verification methodology. This methodology encompasses formal verification, as well as simulation approaches. The methodology enables the definition of formal properties together with an execution control definition to support the definition of an automated verification plan, which links verification definitions and functional requirements.

The shortcomings of current mechatronic development and verification are discussed with respect to verification planning and to current developments in the domain of electronic design. Requirements for a verification plan for mechatronic system verification plan are formulated. Based on these requirements an Enhanced Classification Tree Method is developed, based on the established Classification Tree Method CTM/ES. The new notation and method is embedded into a complete verification plan definition for automatic testbench execution. It supports automatic generation of stimuli, automatic acceptance evaluation and test quality evaluation. A unified notation facilitates horizontal and vertical re-use of descriptions for more efficient definition of a verification plan. The method has been embedded into a current design flow for mechatronic system development.

An exemplary verification plan definition for a modern mechatronic system illustrates the application of the approach, which uses a hardware verification language to define and control a verification environment.

Acknowledgements

This thesis is the result of research performed at C-LAB and the University of Paderborn. Several people supported this research with valuable suggestions and ideas - this is the place for me to thank them.

First of all, I thank Prof. Franz J. Rammig for his guidance during the development of the concepts of this thesis and for asking proper questions at the right time. I thank Prof. Wolfram Hardt for vice-supervising this thesis. I would like to thank Prof. Marco Platzner, Prof. Sybille Hellebrand, and Dr. Peter Pfahler for being members of the examination board.

In C-LAB I have had the opportunity to work on various industrial research projects in the area of electronic and mechatronic system verification, ranging from formal methods to simulation under the auspices of Dr. Wolfgang Müller. I thank him for his advice, his suggestions and for numerous discussions during this time. Not to mention his successful project acquisitions, in some of which I have had the privilege to contribute.

The work of Dr. Mirko Conrad on CTM/ES has been inspiring, and I thank him for his cooperation, beginning with early extensions to CTM/ES. Dr. Stephan Flake has become an asset during my early research on formal methods. The work with Dr. Rafael Radkowski, Henning Zabel, and Henner Vöcking on multiple aspects of mechatronic and embedded system development has always been productive. I thank Prof. Achim Rettberg for discussions about embedded system development, about testing, and for his encouragement on several occasions. Similar thanks go to Dr. Tim Schattkowsky, Dr. Karsten Nebe, Kay Klobedanz.

I thank my family for their continuous support and advice during many of my endeavours, and I thank my friends (you know who you are!) for their patience and for simply being available.

Alfred Alexander Krupp

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Scope	3
1.3	Structure	5
2	Basics and Terms	7
2.1	Terms and Definitions	7
2.2	Verification Technology	8
2.2.1	Static Verification	9
2.2.2	Dynamic Verification	10
2.2.3	Testbench Architecture	13
2.2.4	Stimulus Generation	14
2.2.5	Acceptance Evaluation	15
2.2.6	Test Quality Criteria	16
2.3	Verification Languages	18
2.3.1	Temporal Logic	18
2.3.2	PSL / Sugar	19
2.3.3	<i>e</i> -Language	20
2.3.4	SystemVerilog	21
2.3.5	TTCN	23
2.3.6	Classification Tree Method CTM and CTM/ES	24
2.4	Electronic Design Development and Verification	28
2.4.1	Electronic Design	28
2.4.2	Process	29
2.4.3	Verification Planning	29
3	Mechatronic System Development and Virtual Prototyping	37
3.1	Mechatronic System	37
3.2	Solution Patterns and Partial Models	38
3.3	Process	41
3.4	Virtual Prototyping	42
3.5	Shortcomings of Mechatronic System Verification	43
4	Verification Plan for Mechatronic Systems	45
4.1	Requirements for Systematic Testing of Mechatronic Systems	45
4.2	Concept for Systematic Testing of Mechatronic Systems	47
4.3	Systematic Testing Process	50
4.4	Architecture of Verification Plan and Testbench	52

4.5	Stimulus Patterns for Verification Environment Definition	55
4.5.1	Controllability weaknesses of CTM/ES	56
4.5.2	Enhanced CTM for Stimulus Patterns	59
4.6	Acceptance Criteria for Verification Environment Definition	60
4.6.1	Control characteristics for response evaluation	62
4.6.2	Enhanced CTM for Acceptance Criteria	64
4.7	Test Quality Criteria for Verification Environment Definition	65
4.7.1	Characteristics of Functional Coverage	65
4.7.2	Enhanced CTM for Functional Coverage	68
4.8	Summary: Unified Notation for Systematic Testing Support	70
5	Notation of the Verification Plan Definition and Mapping	73
5.1	Functional Stimulus Definition with Constraints	74
5.1.1	Mapping CTM/ES Stimulus Patterns to Constraints	75
5.1.2	Mapping Enhanced CTM Stimulus Patterns to Constraints	80
5.2	Functional Evaluation Definition – Acceptance Criteria	82
5.2.1	A new notation for CTM Acceptance Criteria Definition	83
5.2.2	Mapping CTM Acceptance Criteria to a Verification Language	88
5.3	Functional Evaluation Definition – Functional Coverage	92
5.3.1	A new notation for CTM Functional Coverage Definition	92
5.3.2	Mapping CTM Functional Coverage to a Verification Language	94
6	Application of the Verification Plan	107
6.1	Example System: RailCab Suspension-Tilt Module	107
6.2	Model-based Test	108
6.2.1	Architecture and Tool Support	109
6.2.2	Formalization of Requirements	111
6.2.3	Creation of Testbench	124
6.2.4	Test Execution	126
6.2.5	Result Evaluation	128
6.3	Towards Physical Test	130
6.4	Summary	133
7	Discussion	135
7.1	Outlook	137